Development of component geographic information systems applying in forest resources management

QI Hong, YU Su-fang, FAN Wen-yi
Forestry college, Northeast Forestry University, Harbin 150040, P. R. China

Abstract: The history, current situation, and development trend of GIS (Geographic Information System) applied in the forest resources management were discussed in this paper. On the basis of geographic spatial characteristics of forest resources data, a component geographic information system (ComGIS) was developed for forest resources management. The system embeds a GIS ActiveX control MapObjects (Inc. ESRI) on Visual C++ platform. System design, data organization and achieving way were studied and expatiated by taking Xigangzi Forestry Centre as study object. The system has many useful functions,, such as adding and display of various map layers, zoom of map by wheeling mouse, attribute and spatial data querying, map auto roaming, features rendering based on the spatial trait of data, label controlling through attribute data band with vector graph, as well as output of 'Column chart' for showing the result of statistics. At the same time, parts of source codes are attached.

Keywords: GIS; ComGIS; Forest resources management; MapObjects; Visual C++
CLC number: P208; S771 **Document code:** A **Article ID:** 1007-662X(2005)01-0047-05

Introduction

Forest resources are work object of forestry department. As a reproducible natural resource, it has some characteristics such as wide distribution, complex terrain, long growth cycle and multiform management object, holding many obvious geographic information characters. It is regional that the data is related with geographic position, and this is the main difference to other information. Due to the limitation of science technology in the past, it can not be treated as spatial data. At present, the management for forest resources mainly adopts the method of managing attribute data of sub-compartment by using computer whereas managing graphical materials manually. It is inevitable that the graphical materials and attribute database can not update synchronously (Hong *et al.* 1991). The birth of Geographic Information System (GIS) provides a powerful tool for the forest resources management.

The definition of GIS is composed of two parts. One side, it is a rising interdisciplinary one that describes, stores, analyses, and outputs spatial information's theory and method; On the other hand, GIS is a computer technology system which is based on spatial database. It provides variety of spatial and dynamic geographic information timely, and works for the geographic study and decision services by using geographic module analytic method (Wu *et al.* 2002). GIS can be looked as a system. It is composed of hardware, software, data, people, institution and scheme which collects, regulates, analysis, distributes regional information of the earth (Dueker 1989). Spatial features are stored in a coordinate system, which references a particular place on the earth. Descriptive attributes in tabular form are associated with spatial features. Spatial data and associated attributes in the same coordinate system can then be layered together for mapping and analysis. GIS can combine spatial and attribute information ideally (Sun *et al.* 2002). This technology is used in many fields to meet variety of needs for application comprehensively. It can do many things from simply map showing and making to geodata' complexly simulate and analysis. The products such as Arc/Info, ArcView, have built the industrial standard for drawing, but during actual work, only parts of functions of GIS are used but supported by huge platform. As a result, large numbers of computer resources are occupied. MapObjects solves the problem. It gives the function of creating and showing maps for development platform, and the programmer can decide the depth of development (Liu 2002).

There are mainly three ways to develop GIS applications as following: designing spatial data structure, database and developing GIS software under Visual C++, Visual Basic, Delphi platform; introducing into GIS software such as ARC/INFO from foreign countries and using provided secondary developed tools; making use of advanced language which support Object-Oriented technology and controls provided by GIS company to fulfill executive applications. This end user-oriented method is called ComGIS (Zhai *et al.*2003). COM is a powerful integrated technology which can connect one software module to another, and the two modules may communicate through interface after connection (Rogerson 1997). One GIS component can communicate with another through standard communication interfaces (Zhaoqiang *et al.* 2004). In this case, MapObjects is chosen as the component. It comprises an ActiveX control (OCX) called the Map control and a set of over forty-five ActiveX Automation objects. There are many advantages by using MapObjects in developing GIS system such as flexibility, efficiency, convenience. By virtue of these advantages, time will be saved and difficulties will be reduced. Creating applications for specific need combining with other applications is what GIS tools are wanting. The maturity of MapObjects surely will push forward the popularization of GIS applications (Chen 2001).

This study employed a new ActiveX Component object technology, realized the micro-core and component structure of system, thus the system has fine extensibility. With the updating of

Foundation item: This work was supported by Provincial Key Technologies R and D program of Heilongjiang (GC02B608)
Biography: Qi Hong (1978-), female, Master student, Forestry college, Northeast Forestry University, Harbin 150040, P. R. China
(E-mail: qhown@sohu.com)
Received date: 2004-10-20
Responsible editor: Chai Ruihai

the system, new component will be added or taken place of old one to achieve the new functions without changing other code (Yang *et. al* 1999). In addition, we have own ourselves copyright. Visual C++ as develop platform was chosen. It is one of the most important application development systems under Windows environment. And GDI (Graphic Device Interface) is the most basic and familiar programming mode. Whenever the program needs to perform screen display and printer plot, GDI functions need to be called. It includes the functions that used to draw point, line, rectangle, polygon, ellipse, bitmap, and text (Hou 2001).

What the forestry department need is a data management system which can manage spatial data and attribute data efficiently. The purpose of the system is using lowest development cost and simplest operation to accomplish the functions, applying GIS to daily work and solving the present problem. The most important thing is that this system can put the operating personnel out of the complex operations.

Data collection

The firsthand data of the system used in this paper are landform maps and five-meter resolution SPOT5 satellite remote sensing images of AiHui District, Heilongjiang Province, China. The maps were made into Shape files through vectorization, and after surveying, attribute database was built.

System design

GIS software engineer applies the idea and method of software engineer in the procedure of GIS software development. For the purpose of improving the changeability of the system, the most efficient way is to utilize modular programming, that is to look the whole system as a modal, then divide it into some sub modals according to functions. One modal execute one function and vice verse. Thus the system can be changeable freely (Liu *et al.* 2004). The system structure is designed as Fig. 1

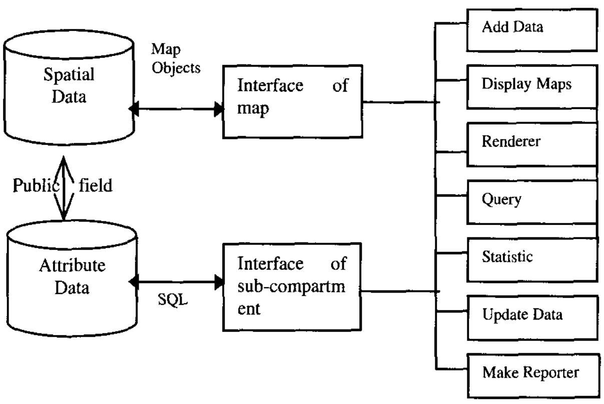

Fig. 1 system structure

The data of a GIS system is divided into spatial data and attribute data. The spatial data comprise of some maps (in this case are forest maps of FUXIN and HONGQI). Each map included some layers such as ROAD, RIVER (see Fig. 2). For organizing attribute data, several tables and procedures were created and put into one database based on the inquisitional data.

Map 1 (HongQi)
Point Layer
Layer 1 (triangular point)
......
Line Layer
Layer 1 (River)
Layer 2 (Road)
Layer 3 (sub-compartment bondary)
Layer 4 (compartment boundary)
......
Polygon Layer
Layer 1 (sub-compartment)
Layer 2 (compartment)
......
Map2 (FuXin)
...

Fig. 2 Organization of spatial data

System functions

Adding MapObjects to Visual C++

The easiest way to use MapObjects in an application is to place the map control on a dialog. The control's properties, methods, and events using the ClassWizard are available. Not only can maps be placed on dialog boxes, also they can be placed in the client area of SDI- or MDI-based applications using MFC's CFormView class. Here, a simple SDI application using Visual C++'s AppWizard was built, and then progressively more advanced map was added to support it (ESRI 2001). Pay more attention to the following steps: "Single Document" for the type of application to create, Select CEasyMapView and make the base class CFormView, copy *mapobjects2.h* from the Common directory into project directory, edit the *stdafx.h* file and add, at the end, *#include "mapobjects2.h"*.

Adding geographic data to the system

MapObjects can mainly use the following files form: Shapefile, Coverage, SDE, VPF, StreetMap and CAD files (Xue 2004)

Shapefile: ESRI Shapefiles use a simple, non-topological format for storing the geometric location and attribute information of geographic features. Shapefiles can be created using some general methods, and the most common way is that they can be created by exporting any data source by using ArcView. In this case, Shapefiles were study objects. It includes road.shp, river.shp, xbm.shp, xbj.shp, etc..

ARC/INFO coverage: ARC/INFO coverage are a topological data structure for geographic features. The coverage format is suitable for spatial analysis and large geographic data management applications.

ImageLayers: In each of the previous types, they are all MapLayer objects based upon vector data sources. MapObjects also allows us to use a wide range of image types as ImageLayers, including such common image types as windows bitmaps (.bmp), tagged image file format (.tiff), and CompuServe bitmaps (.gif).

Displaying map in different forms

Zoom, Full Extent, Pan: The spatial area displayed on the map can be controlled by setting the Extent property. Some simple controls were added that the application will activate in response to mouse clicks inside the map by using the Track Rectangle method to zoom in and zoom out. Write some code that the application can execute in response to the MouseDown event on the map. Then the map will zoom to the full extent. The FullEx-

tent of the Map represents the union of all the MapLayer extents in the Map. The Pan method provides a way to move the Map object's extent with the mouse.

Based on Scale: Sometimes the information on the map is too much to display clearly at the full extent. It can be controlled based on scale. At the full extent, it is not necessary to display overmuch detail; hence in response to the BeforeLayerDraw event, it should selectively make the sub-compartment, compartment visible or invisible, which depends on the current extent of the map. If the width of the current extent is less than or equal to one-fourth of the full extent of the map, then the sub-compartment will be visible and the compartment will be invisible.

Birdseye: Birdseye can provides us the overlook function when the map is too big to be seen conveniently,

```
CMoSymbol sym;
CMoRectangle Rect(m_map.GetExtent());
sym.CreateDispatch(TEXT("MapObjects2.Symbol"));
sym.SetOutlineColor(RGB(255,0,0));
sym.SetStyle(1);
m_smap.DrawShape((LPDISPATCH)Rect,sym);
```

Alter the size of the map when mouse wheel: BOOL CMapObjView::OnMouseWheel (UINT nFlags, short zDelta, CPoint pt)

```
{
    CMoRectangle Rect;
    Rect.CreateDispatch(TEXT("MapObjects2.Rectangle"));
    Rect=m_map.GetExtent();
    if (m_bWheelBig)//if True then zoomin; else zoomout
        Rect.ScaleRectangle(0.8);
    else
        Rect.ScaleRectangle(1.8);
    m_map.SetExtent(Rect);
    return CFormView::OnMouseWheel(nFlags, zDelta, pt);
}
```

Renderer

Each MapLayer object has a Renderer property. A Renderer object controls how MapObjects draws a MapLayer. For showing sub-compartment data clearly, directly, the following functions are added to the system based on the characteristics of attribute data:

1) A LabelRenderer is an object that represents a way of symbolizing features by drawing text on a feature (ESRI 2001). It is a very common way; any of the fields could be used in the record set that stores the text values to use as labels. The Field property is the name of the Field in the Record set that stores the text values to use as labels. The value will be shown in the center of the polygon of sub-compartment. Moreover, in some fields showed compactly at full extent, we can control them based on scale.

```
CMoLabelRenderer lr;
lr.CreateDispatch(TEXT("MapObjects2.LabelRenderer"));
lr.SetField("area");
CMoMapLayer
lyr(m_map.GetLayers().Item(COleVariant("sub-compartment polygon")));
lyr.SetRenderer(lr);
m_map.Refresh();
```

2) A DotDensityRenderer is an object that represents a way of symbolizing features by drawing dots on a feature (ESRI 2001). Each dot represents a uniform, specified value in the Field property of the DotDensityRenderer as specified in its DotValue property. It is fit for the attribute field which own distribute character, such as forest fire towers etc。 Accordingly, it takes convenience to managers that they can contrast each data and get the whole status.

```
CMoMapLayer
lyr(m_map.GetLayers().Item(COleVariant("sub-compartment ")));
CMoDotDensityRenderer ddr;
CMoRecordset recs(lyr.GetRecords());
CMoStatistics stats(recs.CalculateStatistics(TEXT("closure")));
ddr.CreateDispatch(TEXT("MapObjects2.DotDensityRenderer"));
ddr.SetField("closure");
ddr.SetDotValue(stats.GetMax()/100.0);
ddr.SetDrawBackground(TRUE);
lyr.SetRenderer(ddr);
m_map.Refresh();
```

3) A ClassBreaksRenderer is an object that represents a way of classifying features into categories or classes, by drawing different symbols for features (ESRI 2001). The Symbol used to render the feature depends upon the value contained in the specified Field. Take gradient as an example, it was divided to 6 grades, and it became darker and darker with numbers growing (Fig. 3).

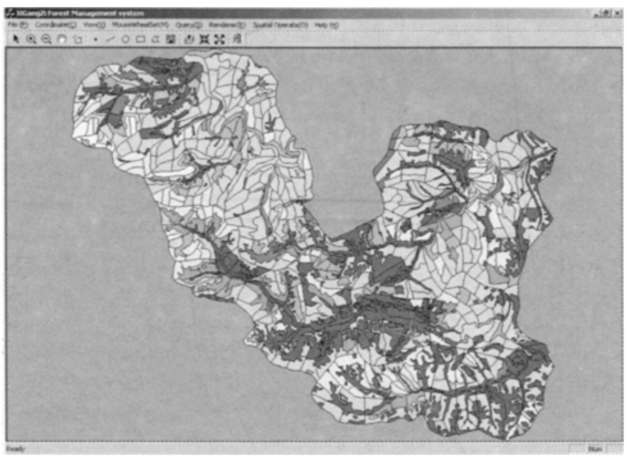

Fig. 3 Result after rendering according to gradient

4) The ChartRenderer Object properties and methods provide the ability to compare multiple attributes of a feature by depicting the attributes as elements of either a pie chart or a bar chart. One feature is used to compare with another by the relative size of each feature's chart and the result is clear at a glance.

There are other render methods can be used too. Here, the above four were discussed, but the accomplished way of them is similar.

Queries

GIS can manage not only spatial data but also attribute data associated with the former, and any one from the other can be searched. On one hand, it implements a simple function for locating a sub-compartment by name. The code first builds a simple SQL query expression using the value of the TextBox, then searches the sub-compartment layer using the SearchExpression method; the result is a Recordset object. In this case, the code got the value of the Shape field for the first record. The code scaled the extent of the shape and then set it to be the extent of the map. The code then redrawed the map explicitly, using the Refresh method; and finally, flashed the shape several times. On the other hand, it can complete feature identification. Click a feature pre-

senting, a sub-compartment in the map, a ListBox would show the information of it. Fig. 4 is a result of query.

In OnMouseMove function:

```
        if (m_curtool==ID_PROPERTYSEARCH)
    {
        m_RsResult=lyr.SearchShape(p2,5,TEXT(""));
        m_RsResult.MoveFirst();
        if (m_RsResult.GetEof())
        {
            return ;
        }
        CString str;
long
nl=m_RsResult.GetFields().Item(COleVariant("age")).GetValue().lVal;
 long
lbh=m_RsResult.GetFields().Item(COleVariant("compartmentnumber")).
GetValue().lVal;
......
long            lzq=m_RsResult.GetFields().Item(COleVariant("forest
type")).GetValue().lVal;
str.Format("age:%d;                  compartmentnumber:%d;
sub-compartmentnumber:%d; land type :%d; grade ofslope:%d; area:%d;
forest type:%d; ",nl,lbh,xbh,tdzl,pdj,bw,area,jycs,lzq,ssmxj,kdxj,spxj);
m_tooltips=str;
UpdateData(FALSE);
VARIANT va;
VariantInit(&va);
va.vt=VT_NULL;
m_bDraw=TRUE;
m_map.GetTrackingLayer().Refresh(true,va);
}
```

In AfterTrackingLayerDraw() function:

```
CMoSymbol sym;
    sym.CreateDispatch(TEXT("MapObjects2.Symbol"));
    CMoPolygon poly;
    CMoField fld;
    sym.SetSymbolType(2);
    sym.SetStyle(0);
    sym.SetColor(RGB(255,0,0));
    if (m_bDraw)
    {
        CMoField
fld=m_RsResult.GetFields().Item(COleVariant(TEXT("Shape")));
        poly.AttachDispatch(fld.GetValue().pdispVal);
        m_map.DrawShape(poly,sym);
        m_bDraw=FALSE;
    }
```

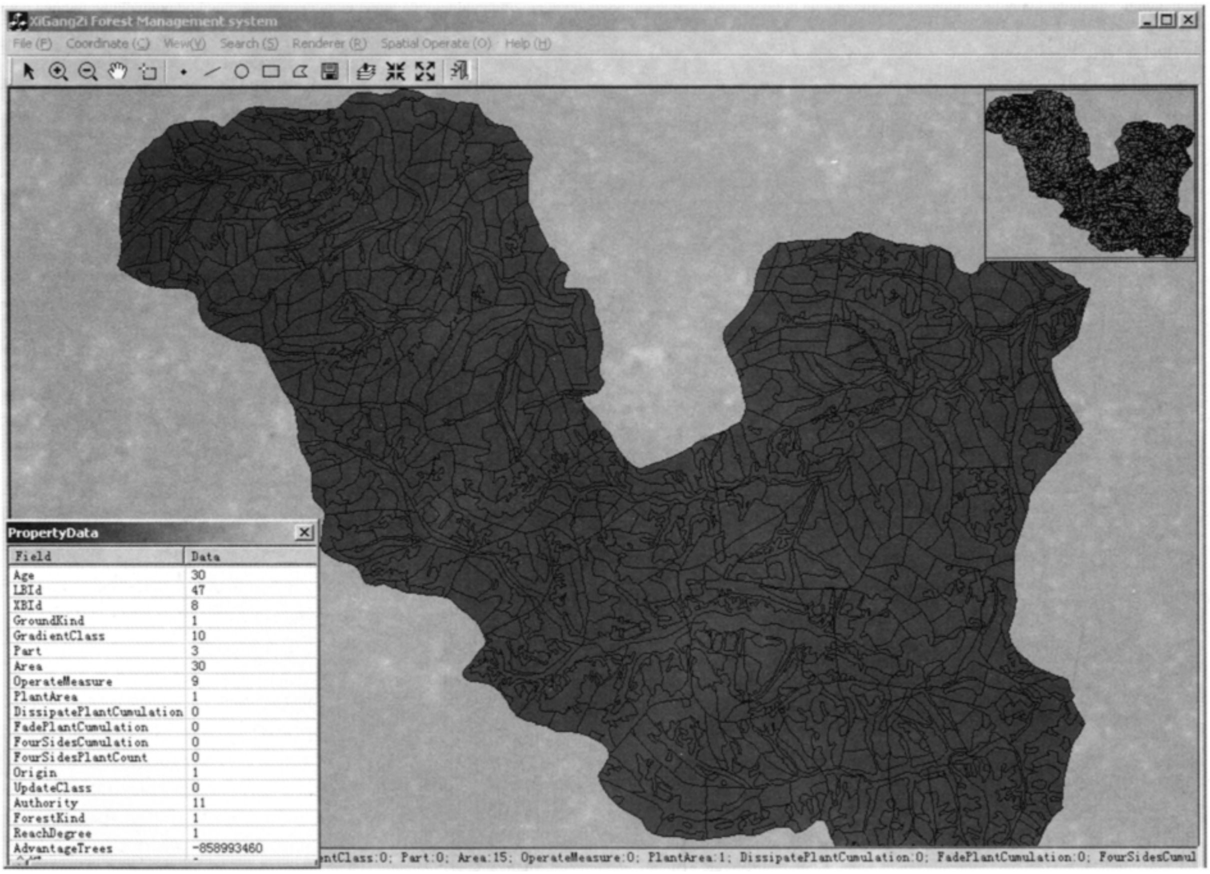

Fig. 4 Result of query corresponding a sub-compartment

Management of attribute data

Statistics are very important in forestry application. In this case, the sum of area, cumulation of sub-compartment etc. were counted from original database. OLE (Object Linking and Embedding) technology was used, and version 10 of Cristal Re-

porter is imported in develop environment to design the reporter; connection was build with database through ADO (ActiveX Data Objectcs) reference (Dong *et. al* 2003); Some Procedures in SQL Server were created to fulfill the insert, delete, update, query functions. At last, the result was shown to the user by form of 'Column chart' or 'Pie chart' and could be printed. We take age for example: the range of age is from 0 to 4, the system provided 'Column chart' to show the sum of area of each age (Fig. 5).

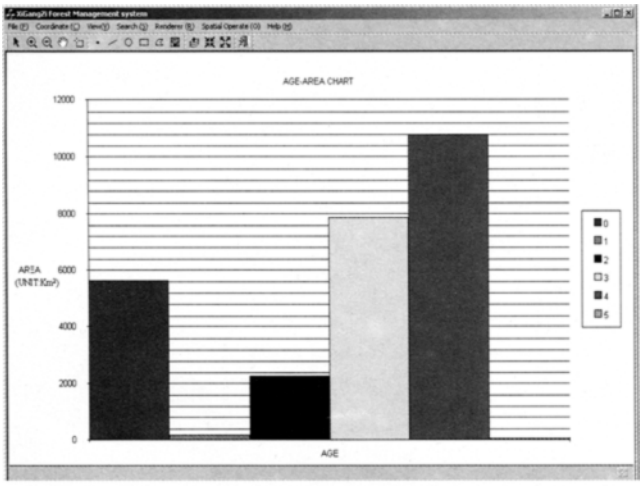

Fig. 5 Result of statistic based on age

Spatial data and attribute data update synchronously

The data will change according to requirement. Therefore, the function that spatial data and attribute data update synchronously is especially important. In this system, if spatial data altered, the attribute data associated with it would also change through keywords and vice versa. The system provided the function of create and edit geometric objects like POINT, LINE, RECTANGLE, ELLIPSE, etc.. After modifying the graph of sub-compartment by using them, write the information into database at the same time.

Conclusion

In this case, Visual C++ platform embedding MapObjects was adopted. It showed the obvious advantage of VC++ mainly because it can use GDI (Graphical Device Interface) to operate graph and image efficiently, and also, it can call API (Application Program Interface) functions directly. It brought us more conveniences and much higher programming speed. Moreover, it also provides DLL (dynamic Linking Library) Wizard, and supports regular written DLL, so the extendable capabilities of the system were enhanced.

After developing, the notable characteristics of GomGIS were found. It is easier to control the program by writing a few codes instead of learning a new develop language. Programmers can add map control to the application, and user end-oriented applications could be built and contacted with the requisite information.

Many objects, properties, methods of MapObjects were used in this case, and Rendering is worth mentioning. By using some Renderer objects, the suitable way according to the data's traits in concrete appliance could be chosen. This gave the users a clear optical effect and provided a basis for the next decision.

Through earnest study, this system has accomplished parts of the functions of Forest Resource Management system. But due to the limitation of MapObjects, it can't executive some senior functions like high quality map output, senior spatial analysis and topological edit (surface model or network analysis), applicators need to choose ESRI's other products to carry out them.

Acknowledgments

We thank Prof. Li Feng-ri from Northeast Forestry University for his valuable comments. We are grateful to Wang Chao and Zhou Feng from Harbin Gold Star Electronic Company for their help in solving some technical problems. Thank Sun Lu and Guo Hui for their hard work in reviewing the manuscript.

References

Chen, Y. 2001. Geographic information system design and development based on MapObjects [J]. Computer Engineering, **27**(4): 150–152. (in Chinese)

Dong, R., Yu, F. 2003. Using ADO to Manage SQL Server Database in VC [J]. Applied Science and Technology, **30**(2):25–27.

Dueker K.J, Kjerne, D. 1989. Multipurpose cadastre: terms and definitions. ACSM-ASPRS. Washington. DC: Annual Convention, pp.94–103.

ESRI. 2001. Getting Started with MapObjects Version 2.1 in Visual C++

ESRI. 2001. MapObjects2.1 Online Reference

Hong, L.X, Tang, S.Z, Cai, D.X. 1991. Silvicultural and forest resource information management system in Daqingshan Experiment Bureau [J]. Forest Research, **4**(Supp.):22–31. (in Chinese)

Hou, J.J. 2001. Dissecting MFC 2/e [M]. Wuhan: Huazhong University of Science & Technology Press, p337–339. (in Chinese)

Liu, G., Liu, X.D. 2004. Geographic information system secondary developed samples course—VB.net and MapObjects realization [M]. Beijing: Tsinghua University Press, p2–3. (in Chinese)

Liu, Y.M. 2002. MapObjects—a Flexible Development Tool of GIS [J]. Geology of Anhui, **12**(3):208–214. (in Chinese)

Rogerson, D. 1997. Inside COM. Washington: Microsoft Press, p1–293.

Sun, Y.D, Liu, G. 2002. The application of MO(MapObjects) in the development of field information system [J]. Journal of China Agricultural University, **7**(6): 72–77. (in Chinese)

Wu, L., Liu,Y., Zhang, J., Ma, X.J. 2002. Geographic information system: Principles, methods and applications [M]. Beijing. Science Press, p6–8. (in Chinese)

Xue, W. 2004. MapObjects—Geographic information system program design [M]. Beijing: National University of Defense Technology Press, p3–5. (in Chinese)

Yang, D.Q, Jiang, N. 1999. WUXI sustainable development decision supporting system based on MapObjects [J]. Map, 2: 19-26. (in Chinese)

Zhai, C.J, Zhang, Q.W. 2003. Developing GIS application system with activeX components MapObjects [J]. Microcomputer World, **19**(5): 82-83. (in Chinese)

Zhao, Q., Gong, H.L, Zhao, W.J, Deng, W. 2004. Implementation of the fundamental Capabilities in ComGIS Based on VC++ [J]. Geo-Information Science, **6**(1): 41-43. (in Chinese)